

Review dan Analisa Faktor-Faktor yang Mempengaruhi Kecepatan Akses Halaman Website

Himawan

Teknik Informatika, STMIK Raharja, Tangerang, Indonesia

himawanawan10@gmail.com

Diterima 15 maret 2019

Disetujui 24 Juni 2019

Abstract— The change in the behavior of internet users from using computers or laptops to mobile internet users makes changes in the way the browser and also the web pages display information. Internet users generally want a quick access time when visiting a website page to get the desired information. In the research conducted in the writing of this journal, the researchers wanted to show and explain the several important factors that influence the speed of access from a website page, as well as analyzing based on technical factors. Where the main discussion in this study will focus more on the evaluation of technical factors starting from the programming side (server side programming and client side programming) and also the design of the user interface using web pages using minify CSS along with the use of AJAX technology. The results to be achieved from this study are to identify how much influence the technical factors mentioned above have on the speed of visitor access to a web page, apart from other technical factors such as internet network speed, devices and areas where users can access website page.

Index Terms—Access Speed, AJAX, Website Access Time, Minify CSS, Website Programming

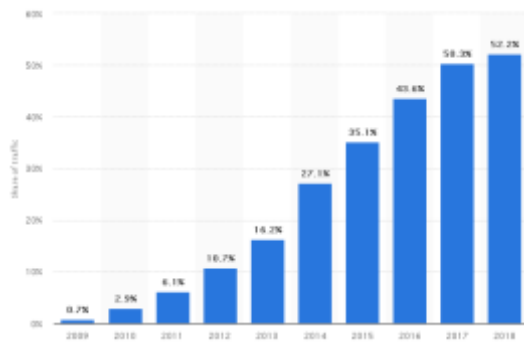
I. PENDAHULUAN

Era awal penggunaan halaman *website* pada tahun 1990, mulanya hanya digunakan untuk kebutuhan yang sangat sederhana yaitu halaman-halaman *website* statis yang hanya menampilkan informasi berupa teks. Dimana halaman *website* pertama kali dibuat oleh Tim Berners-Lee yang merupakan “Bapak *Website*” atau penemu dari WWW (*World Wide Web*) melakukan riset di Laboratorium CERN (Organisasi Penelitian Nuklir Eropa) [1]. Halaman *website* statis tersebut hanya berisikan *script* atau kode HTML (*Hyper Text Markup Language*) merupakan versi awal dari teknologi halaman *website* atau yang akan dikenal dikemudian hari dengan istilah web 1.0 [2].

Halaman *website* pada era web 1.0 yang hanya menggunakan kode atau *script* HTML tidak mengenal istilah “*syntax error*”, namun tidak demikian halnya dengan penggunaan bahasa pemrograman web *server* yang memiliki aturan lebih ketat dan juga mengenal adanya *error* atau kesalahan pada penulisan kode program. Selain itu, untuk membuat halaman *website*

menjadi lebih interaktif juga diperlukan teknologi pemrograman dari sisi klien (*client side programming*) yaitu dengan menggunakan JavaScript. Secara garis besar dari penjelasan pada paragraf ini yaitu halaman *website* yang dilihat oleh user merupakan perpaduan dari beberapa teknologi yang digunakan untuk membuat halaman *website*. Dengan demikian waktu akses terhadap halaman *website* juga akan meningkat karena perpaduan teknologi tersebut dan juga ditambah dengan komponen-komponen pendukung lainnya seperti penggunaan gambar (*image*) dan video yang ditampilkan pada sebuah halaman *website*.

Masalah mengenai kecepatan akses halaman *website* sebenarnya bukan merupakan hal yang baru pada era teknologi informasi dan komunikasi digital. Hal ini telah menjadi perhatian dan disadari sejak era millenium dan banyak pengembang *website* telah berusaha untuk mengoptimalkan teknik-teknik optimasi dalam pemrograman web dan juga sistem basis data (*database*) untuk mengurangi beban kinerja dari web server dan resource *server* itu sendiri. Kemudian, kecepatan akses halaman *website* menjadi sangat krusial, terutama jika dilihat trend dalam 5 tahun terakhir dimana jumlah pengakses halaman *website* juga sudah mengalami pergeseran yang signifikan yaitu para user atau pengguna cenderung menggunakan *smartphone* untuk berbagai aktifitas kegiatan dalam berselancar di dunia maya (*surfing*). Hal ini sangat berbeda jika dilihat sebelum tahun 2010, dimana pengguna internet lebih banyak menggunakan komputer pc (desktop) ataupun laptop untuk mengakses halaman *website*. Pada gambar 1 berikut merupakan grafik yang menunjukkan pertumbuhan pengguna internet mobile yang mengalami peningkatan setiap tahunnya seperti yang dilansir oleh peneliti dari halaman *website* statista.com.



Gambar 1. Statistik Penggunaan Smartphone Untuk Mengakses Halaman Website

Pembahasan pada penelitian dan penulisan jurnal ini hanya akan membahas beberapa faktor kunci mengenai hal-hal yang mempengaruhi kecepatan akses halaman *website* sebagai berikut :

1. Optimasi teknik pemrograman web *server* dan dalam hal ini, bahasa pemrograman yang akan dibahas adalah pemrograman PHP. Alasan utama dari pemilihan pemrograman PHP adalah karena tidak lain karena penggunaan pemrograman PHP sangatlah dominan jika dibandingkan dengan bahasa pemrograman web lainnya seperti JSP atau ASP [3]. Selain itu variasi dari framework PHP yang lebih banyak dan beragam juga menjadi alasan kuat mengapa pemrograman PHP juga menjadi bahasa pemrograman web paling populer digunakan [4].
2. Penggunaan JavaScript yang merupakan bahasa pemrograman berbasis *client side scripting* untuk web yang paling populer. Dimana pengembangan JavaScript menghasilkan beberapa library baru yaitu JQuery dan juga Framework Bootstrap.
3. Penggunaan CSS (*Cascading Style Sheet*) yang digunakan untuk “mempercantik” tampilan atau layout dari desain antar muka (*user interface*) halaman *website*.
4. Penggunaan teknologi AJAX, namun sebelumnya kenapa poin AJAX dijadikan terpisah dan tidak menjadi satu dengan penggunaan JavaScript?. Jawabannya adalah karena penggunaan JavaScript secara mandiri (*stand alone*) tidak dapat melakukan proses transfer dari klien dan server secara asynchronous.
5. Sisi teknis mengenai penggunaan jaringan internet (kualitas sinyal atau jenis jaringan) dan berbagai penggunaan perangkat keras dalam jaringan internet seperti jenis modem atau router tidak akan dibahas pada penelitian ini.

Hal utama yang ingin dicapai pada penelitian ini adalah terkait dengan bagaimanakah cara atau teknik yang dapat dilakukan untuk mengurangi waktu akses pada sebuah halaman *website* dengan menggunakan teknik optimasi pemrograman web server, memperkecil ukuran file JavaScript dengan teknik kompresi (*minify JavaScript*), memperkecil ukuran file CSS (*minify CSS*) dan terakhir adalah penggunaan Teknologi AJAX. Selain itu penelitian ini juga bertujuan untuk meningkatkan dan mengoptimalkan kinerja dari halaman *website* dengan meminimalkan kompleksitas dari beberapa faktor kunci yang telah dituliskan dan dijelaskan sebelumnya. Kemudian terakhir adalah melihat hasil dari pengujian optimasi halaman *website* yang diukur dengan menggunakan *web page tool analyzer*.

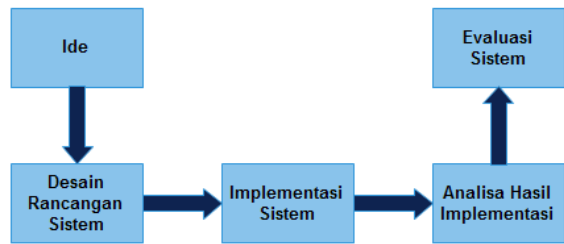
II. METODE PENELITIAN

Metode penelitian memegang peranan yang sangat penting dari proses sebuah penelitian untuk menghasilkan penelitian yang berkualitas [5]. Penelitian yang dilakukan oleh peneliti dalam penulisan jurnal ini akan menggunakan 2 buah metode penelitian yaitu menggunakan metode eksperimen dan juga metode literature review. Kedua metode tersebut dipilih berdasarkan pertimbangan faktor-faktor berikut :

1. Metode eksperimen dipilih berdasarkan fakta bahwa metode ini merupakan salah satu metode paling populer digunakan dalam bidang penelitian ilmu komputer [6]. Selain itu penggunaan metode eksperimen sangat bermanfaat untuk melakukan serangkaian tahapan pengujian pada perangkat lunak ataupun sistem untuk mendapatkan hasil yang optimal.
2. Metode Literature Review digunakan pada penelitian ini sebagai dasar bahan studi perbandingan antara penelitian sekarang dengan penelitian terdahulu. Dimana metode literature review akan dapat dengan jelas memperlihatkan posisi atau perbedaan antara 1 penelitian dengan penelitian yang lainnya.

A. Metode Eksperimen

Penggunaan metode eksperimen pada penelitian dan penulisan jurnal ini bertujuan untuk melakukan pengujian dan juga pembuktian bahwa faktor-faktor kunci yang mempengaruhi kecepatan akses halaman *website* adalah benar dan valid. Kemudian hasil dari eksperimen yang telah dilakukan akan ditunjukkan dalam bentuk visualisasi grafik ataupun tabel perbandingan uji evaluasi.



Gambar 2. Alur Metode Penelitian Menggunakan Model Eksperimental

Sebagai contoh, salah satu model desain rancangan sistem yang diusulkan pada penelitian ini adalah teknik penyederhanaan dari penulisan script CSS atau yang juga dikenal dengan istilah *minify* CSS. Dimana akan terdapat beberapa file CSS yang akan dikompres dengan menggunakan bantuan tools online seperti css minifier atau css compressor. Kemudian hasil dari kompresi file css akan dibandingkan dengan file css yang asli sebelum dikompres untuk melihat seberapa signifikannya waktu akses halaman website terhadap file css yang telah di *upload* ke dalam web server untuk dieksekusi oleh *browser* pada saat halaman website diakses atau dikunjungi oleh user. Pada gambar 3 berikut ini adalah proses yang menunjukkan penyederhanaan atau minify dari script CSS yang telah disederhanakan namun tanpa merubah script aslinya. Dimana ukuran asli script CSS sebelum diminify adalah sebesar **387 byte** dan setelah dilakukan proses minify menjadi **231 byte**. Terdapat perbedaan selisih ukuran sebesar **156 byte**. Rasio kompresi ukuran file CSS setelah dilakukan proses kompresi adalah sebesar **59.68 persen**.



Gambar 3. Proses Kompresi atau Minify Script CSS

B. Literatur Review

Peneliti mengangkat topik mengenai pembahasan kecepatan akses pada halaman website karena topik ini merupakan salah satu topik yang menarik untuk dibahas karena salah satunya adalah mengenai faktor kepuasan pengunjung atau user yang mengakses sebuah halaman website. Banyak penelitian dan juga survey yang membuktikan bahwa kecepatan akses halaman website sangat berpengaruh pada tingkat popularitas atau ranking dari halaman website. Berikut ini adalah beberapa penelitian yang dijadikan sebagai bahan referensi atau studi perbandingan :

1. Manhas [7] menuliskan dalam papernya mengenai analisa faktor-faktor yang dapat

meningkatkan kecepatan akses halaman website. Penelitian ini menekankan pada teknik optimasi meningkatkan kinerja halaman website dengan cara mengurangi atau memperkecil ukuran file HTML pada setiap halaman website, kemudian teknik kompresi image, teknik CSS validation dan HTTP Caching.

2. Sakamoto *et al* [8] membahas bahwa penyederhanaan bentuk penulisan kode program JavaScript dari bentuk raw script file menjadi bentuk yang lebih minimalis (*minify script*) akan menghemat waktu transfer data halaman website dan sekaligus juga akan mengurangi beban dari lalu lintas (*traffic*) yang terjadi pada jaringan komputer. Selain itu pada penelitiannya juga membahas mengenai teknik HTTP compression yang merupakan teknik untuk mengurangi jumlah dari permintaan yang dilakukan oleh user terhadap konten dari sebuah halaman website, dimana teknik HTTP compression yang dibahas pada penelitiannya menggunakan algoritma kompresi Lempel-Ziv (LZ77) dan dikombinasikan dengan algoritma Huffman. Penelitian ini menjawab akan 2 hal penting yaitu : berapa banyak waktu akses website yang dapat dihemat atau menjadi berkurang setelah proses optimasi dilakukan dan hal yang kedua adalah berapa besar prosentase pengurangan ukuran file JavaScript yang disisipkan atau diletakkan pada halaman website.

3. SyamimiSaid *et al* [9] membahas dalam penelitiannya bahwa performa atau kinerja dari halaman website merupakan bagian yang sangat penting dari layanan jaringan internet. Pada penelitian ini juga melakukan survey terhadap faktor-faktor penting yang mempengaruhi kecepatan akses halaman website, dimana faktor-faktor tersebut diantaranya adalah ukuran halaman website yang terlalu besar, komparabilitas browser dalam menangani halaman website (cara kerja setiap browser berbeda), kemudian yang terakhir adalah faktor user atau pengunjung web itu sendiri. Hasil dari penelitian ini adalah survey mengenai teknik-teknik yang digunakan untuk meningkatkan kinerja dari halaman website dengan cara teknik request HTTP, minifying CSS HTML dan JavaScript, kemudian yang terakhir adalah kompresi image.

III. HASIL DAN PEMBAHASAN

Pembahasan utama penelitian ini akan dibagi ke dalam 4 sub-bab (*section*) sesuai dengan batasan atau ruang lingkup yang telah dituliskan pada bagian awal pendahuluan yaitu optimasi pemrograman web server, optimasi penggunaan JavaScript, teknik minify CSS dan yang terakhir adalah penggunaan AJAX. Kemudian hasil dari pengujian dengan menggunakan metode eksperimen akan ditampilkan dalam bentuk tabel dan juga grafik batang. Hal ini perlu dilakukan untuk melihat seberapa besar pengaruh dari penggunaan teknik optimasi yang digunakan pada penelitian dan penulisan jurnal ini.

Jumlah file (*script source program*) yang akan digunakan sebagai bahan uji eksperimen pada pembahasan ini berjumlah 7 file dengan rincian : 3 file php, 2 file css, dan 2 file javascript. Berikut adalah print screen dari file yang dimaksudkan untuk menghindari manipulasi ukuran data file yang dimaksudkan. Kemudian untuk nama file program php juga telah disamarkan dengan alasan kerahasiaan data source program.

Name	Type	Size
mos-style	Cascading Style Sheet Document	18 KB
style	Cascading Style Sheet Document	6 KB
script1	JScript Script File	58 KB
script2	JScript Script File	98 KB
datauji_1	PHP File	7 KB
datauji_2	PHP File	6 KB
index	PHP File	6 KB

Gambar 4. File Script Untuk Pengujian Eksperimen

Pemilihan 7 script yang ditampilkan pada gambar 4, berdasarkan kondisi dimana file-file tersebut adalah file-file yang sering diakses dan akan digunakan dalam aplikasi sistem yang dilakukan uji eksperimen. Contohnya 2 file css yaitu style.css dan mos-style.css merupakan 2 file css utama yang hampir selalu disertakan (*include*) dalam setiap tampilan halaman website (*dissertakan dalam script HTML*). Jadi pada hasil akhir pembahasan akan dapat terlihat seberapa besar pengaruh pengurangan waktu yang terjadi setelah script atau file css dilakukan proses *minify*. Akankah pengaruhnya signifikan atau bahkan tidak ada pengaruhnya sama sekali terhadap kecepatan waktu akses ?. Hal ini juga akan dibuktikan dengan menggunakan tabel perbandingan waktu akses dari hasil proses uji eksperimen dan simulasi.

A. Optimasi Pemrograman Web Server

Penggunaan teknik optimasi pemrograman web server menjadi sangat penting untuk dilakukan, hal ini tidak lain dikarenakan bahasa pemrograman web server menjadi media perantara atau penghubung antara bahasa scripting (*client side programming*) dengan web server. Oleh karena itu optimasi pemrograman web server khususnya pada penelitian

ini mengenai teknik pemrograman php merupakan hal yang paling utama dan penting. Berikut ini adalah 2 cara sederhana namun sering terabaikan oleh banyak programmer PHP, namun 2 cara ini dapat meningkatkan kinerja dari program php yang digunakan dalam sebuah website yaitu :

1. Teknik menulis kode program PHP secara optimal. Beberapa cara diantaranya adalah :
 - a. Mengoptimalkan penggunaan fungsi looping (fungsi *for* atau *while*).
 - b. Eliminasi atau hilangkan penggunaan fungsi yang berulang, dimana dalam fungsi tersebut juga menggunakan parameter yang sama.
 - c. Hindari penggunaan komentar yang berlebihan pada source program, contoh praktik sederhananya adalah tidak perlu setiap baris program diberikan komentar, walaupun komentar sangat bermanfaat untuk dokumentasi atau “peringat” bagi programmer.
 - d. Penggunaan array multi-dimensi juga untuk sedapat mungkin dihindari. Jika data tersebut masih dapat disimpan dalam bentuk array 1 dimensi.
2. *Tweaking tools* untuk mendapatkan nilai atau *score benchmark* dari hasil tunning kode program yang telah selesai dibuat dan tidak terdapat kesalahan penulisan di dalamnya.

Beberapa poin-poin yang telah dituliskan dan dijelaskan diatas mungkin terlihat seperti “tutorial” atau tips bagi para programmer PHP. Namun, faktanya hal yang terlihat “sepele” atau dipandang remeh dapat berdampak besar aplikasi website yang sudah kompleks. Penulisan kode program PHP yang optimal akan berpengaruh terhadap kinerja dari web server yang akan memproses setiap program php yang akan dieksekusi. Selain itu, ini adalah penelitian yang dituliskan dalam bentuk paper atau jurnal yang memiliki “keterbatasan” dalam hal jumlah halaman dan bukan dalam bentuk buku yang dapat memiliki jumlah halaman hingga ratusan lembar. Oleh karena itu, berikut adalah 1 contoh sederhana yang konkrit untuk teknik penulisan program PHP yang optimal.

```

Model fungsi perulangan 1 :
for ( $i=0; $i<count($data); ++$i )
//statement

Model fungsi perulangan 2 :
$panjangarray = count($data);
for ( $i=0; $i<$panjangarray; ++$i )
//statement

```

Contoh sederhana dari penulisan program diatas dapat dilihat dengan mudah, model penulisan pertama memang hanya membutuhkan 1 baris kode program (*line of code*) jika dibandingkan dengan model yang ke-2 yang memerlukan 2 baris kode program. Secara “logika” sederhana, maka web server akan

mengeksekusi model penulisan pertama dengan lebih cepat bukan ?. Namun kenyataannya atau faktanya adalah tidak seperti itu, kenapa hal tersebut bisa terjadi ?. Jawabannya adalah karena model penulisan fungsi perulangan yang pertama, akan mengeksekusi atau membuat fungsi count(\$data) untuk setiap iterasi atau perulangan. Hal yang akan sangat berbeda untuk fungsi perulangan pada model 2 yaitu pemanggilan dan pembuatan fungsi count(\$data) **hanya dilakukan 1 kali** yaitu pada saat inisialisasi variabel \$panjangarray dibuat. Pertanyaan sederhananya adalah bagaimana jika terjadi proses perulangan sebanyak 100 kali ?. Jika kedua model program fungsi perulangan tersebut dijalankan atau diimplementasikan ke dalam sebuah aplikasi atau sistem ?. Siapakah pemenangnya ?. Jawabannya adalah bentuk model penulisan yang kedua akan menjadi pemenang dalam hal kecepatan proses eksekusi dari awal sampai dengan selesai.

B. Optimasi JavaScript

Teknik optimasi pemrograman JavaScript yang akan dibahas dan digunakan pada penelitian ini adalah dengan menggunakan teknik *concatenation* (penggabungan) dan kompresi file JavaScript (JavaScript Compressor). Untuk penggunaan teknik concatenation file atau source program JavaScript, maka diperlukan juga teknik pemrograman dari sisi server (server-side). Hasil akhir yang ingin dicapai dari teknik penggabungan ini adalah teknik pemanggilan semua file JavaScript dengan menggunakan pemrograman PHP. Berikut adalah contoh sederhana atau yang umum dilakukan oleh banyak programmer web ketika ingin memanggil file JavaScript pada halaman website.

```
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript" src="js-script.js"></script>
<script type="text/javascript" src="data.js"></script>
<script type="text/javascript" src="validasi.js"></script>
</head>
```

Berdasarkan contoh penulisan diatas, dapat dilihat bahwa jika terdapat 4 file JavaScript, maka cara yang diperlukan untuk memanggil 4 file tersebut adalah dengan memanggilnya satu persatu dan menyisipkannya ke dalam tag <head>. Namun cara ini akan membutuhkan waktu pemanggilan atau load file, karena file JavaScript akan dicari terlebih dahulu keberadaannya dan kemudian di eksekusi oleh browser. Cara diatas dapat diubah dengan menggunakan teknik *server-side programming* untuk meminimalkan waktu pemanggilan file JavaScript.

```
//Penggunaan Fungsi Penggabungan File JavaScript
function GabungFile($filejs){
    $bukafile = fopen($filejs, "r");
    if ($bukafile != null){
        echo fread($bukafile, filesize($filejs));
    }
}

//Pemanggilan Fungsi Penggabungan File
GabungFile("jquery.js");
GabungFile("js-script.js ");
GabungFile("data.js ");
GabungFile("validasi.js ");
```

Teknik kedua yang digunakan untuk mempercepat akses halaman website pada pemrograman JavaScript adalah dengan menggunakan teknik kompresi file JavaScript (JavaScript Compression). Hal ini dilakukan untuk memperkecil ukuran (size) dari file source program JavaScript ketika akan dipanggil dan dieksekusi oleh browser. Penggunaan teknik JavaScript Compression pada file JavaScript yang memiliki ukuran besar (50-100 kb) akan memiliki dampak yang cukup signifikan terhadap hasil akhir ukuran kompresinya. Hal ini dapat dibuktikan pada hasil optimasi dengan menggunakan teknik kompresi pada 2 contoh file script yang dijadikan sebagai contoh pengujian yaitu script1.js dan script2.js seperti yang ditunjukkan pada gambar 4. Berikut adalah Tabel yang menunjukkan hasil perbedaan ukuran (size) dari script1.js dan script2.js yang telah dikompresi dengan menggunakan bantuan tools online jscompress :

Tabel 1. Perbandingan Ukuran File JavaScript

Nama file	Ukuran asli (kb)	Ukuran Hasil Kompresi (kb)	Rasio Kompresi (dalam %)
script1.js	58.00	27.79	52.92
script2.js	97.80	45.11	53.85

Hal terakhir yang juga sedikit berpengaruh pada pembahasan mengenai optimasi pemrograman JavaScript adalah posisi atau peletakkan source dari JavaScript, baik penulisan dengan cara menyisipkan source JavaScript secara langsung dalam script HTML ataupun menggunakan file eksternal. Penelitian yang dilakukan oleh penulis Sawant [10] dan juga peneliti Munyaradzi [11] menjelaskan bahwa memindahkan source JavaScript dari cara tradisional yaitu didalam tag <head> menjadi setelah tag </html> akan dapat membuat halaman website menjadi lebih ringan dan cepat untuk dieksekusi oleh browser, hal ini dikarenakan halaman website akan diload terlebih dahulu sebelum melakukan pemanggilan terhadap source program JavaScript yang disertakan atau disisipkan dalam script HTML.

Penggunaan teknik kompresi source program JavaScript memang terbukti dapat membuat ukuran source program JavaScript menjadi lebih kecil, namun teknik ini juga memiliki kelemahan atau

kekurangan didalamnya yaitu source program JavaScript menjadi sulit untuk “dibaca” dan dimengerti oleh programmer. Hal ini akan berakibat proses *debugging* source program akan menjadi lebih sulit karena struktur penulisan source program telah berubah dan “tidak teratur”. Hal ini telah dijelaskan pada penelitian yang dilakukan oleh Mjelde [12] bahwa fitur *source mapping* yang ada pada browser-browser modern akan membantu mengembalikan bentuk atau struktur asli dari program JavaScript sebelum dilakukan proses kompresi, sehingga memudahkan programmer untuk memodifikasi source program dari JavaScript untuk dapat diupdate kembali.

C. Optimasi CSS (Cascading Style Sheet)

Penggunaan CSS dalam desain halaman website, dapat dianalogikan seperti ilustrasi tembok atau dinding tanpa cat. Karena dengan penggunaan CSS akan membuat halaman website menjadi lebih indah dan menarik untuk dilihat. Namun sayangnya, penggunaan CSS juga merupakan salah satu faktor yang menjadikan halaman website memerlukan waktu lebih lama untuk di-load dan ditampilkan seutuhnya. Terdapat beberapa faktor dalam penggunaan CSS yang dapat menjadikan hal tersebut terjadi, diantaranya adalah sebagai berikut :

1. Selector yang ada dalam script CSS tidak pernah digunakan atau dipanggil.
2. Penggunaan property dan property value yang memiliki nilai yang sama. Contohnya property untuk warna dan jenis font yang digunakan.
3. Terdapat pengulangan penulisan baik dalam selector, .id dan #class pada script css sehingga ukuran file script CSS bertambah.

Karena penggunaan CSS hampir tidak dapat dipisahkan dengan desain halaman website, walaupun terdapat alternatif lain yaitu dengan penggunaan template atau desain yang dibuat dengan menggunakan software seperti Photoshop. Namun tetap saja, fakta bahwa CSS sulit tergantikan karena fleksibilitasnya dalam fungsinya untuk mempercantik dan memperindah tampilan dari halaman website. Contoh script CSS pada Tabel 2 berikut adalah untuk menunjukkan ukuran file dari hasil kompresi file CSS dengan menggunakan program atau tools kompresi CSS online yaitu *csscompressor*.

Tabel 2. Perbandingan Ukuran File CSS

Nama file	Ukuran asli (bytes)	Ukuran Hasil Kompresi (bytes)	Rasio Kompresi (dalam %)
mos-	18242	16295	11.95
style.css	5629	3686	52.71

Perbedaan yang terdapat antara teknik kompresi pada JavaScript dengan kompresi CSS, berdasarkan

perbandingan yang dilihat oleh peneliti pada masing-masing tools yang digunakan adalah sebagai berikut :

1. Pada teknik kompresi CSS terdapat opsi tambahan untuk memperkecil ukuran file CSS yaitu penyederhaan atribut (property) untuk warna, contohnya jika nilai atribut warna yang digunakan adalah putih (`#ffffff`) maka penulisan kode hexadecimal untuk warna putih dapat diringkas menjadi `#fff` sehingga dapat menghemat *space* sebesar **3 bytes**. Namun teknik ini hanya bisa digunakan pada warna-warna primer dan tidak dapat digunakan pada warna-warna sekunder seperti contohnya magenta (`#ff00ff`).
2. Teknik kompresi yang digunakan pada file CSS juga memiliki beberapa opsi tambahan lain untuk lebih memperkecil ukuran dari file CSS, namun tidak mempengaruhi sama sekali dari fungsinya. Contohnya adalah menghilangkan penggunaan tanda backslash (\) yang tidak diperlukan, kemudian kompresi dari properti font-weight.

D. Penggunaan AJAX (Asynchronous JavaScript and XML)

Pembahasan terakhir terhadap faktor-faktor yang mempengaruhi kecepatan akses pada halaman website adalah penggunaan teknologi AJAX. Dimana penggunaan teknologi AJAX merupakan hasil pengembangan dari teknologi API (Application Programming Interface) yang dikenal dengan nama DOM (Document Object Model). Teknologi API sendiri diperkenalkan atau digunakan pada tahun 1995, ketika browser Netscape versi 2 diluncurkan untuk membuat halaman website menjadi lebih interaktif bagi pada penggunaanya [13].

Penggunaan teknologi AJAX pada halaman website, memungkinkan untuk terjadinya proses komunikasi dan transfer data antara klien dan server secara asynchronous, hal ini dapat membuat proses load halaman website menjadi lebih cepat jika dibandingkan dengan model halaman website yang tidak menggunakan AJAX [14]. Penelitian yang dilakukan oleh Himawan [14], membuktikan bahwa dengan penggunaan teknologi AJAX dapat menghemat penggunaan bandwidth jaringan internet, jika terjadinya lonjakan permintaan (request) yang dilakukan oleh para pengguna yang mengakses halaman website tertentu. Hal ini bisa terjadi karena pada teknologi AJAX terdapat fungsi XMLHttpRequest, sehingga halaman website tidak perlu melakukan proses “refresh” atau load ulang untuk setiap permintaan (request) baru dari user. Hal ini sangat berbeda jika dilihat pada model halaman website tradisional yang akan melakukan proses “load” untuk setiap permintaan dari user, contohnya adalah proses simpan data ke dalam database server

atau pada saat meng-klik link pada sebuah halaman website.

E. Hasil Optimasi Secara Keseluruhan

Pembahasan pada sub-bab ini adalah mengenai hasil dari pengujian menggunakan eksperimen dari faktor-faktor yang telah dianalisa dan dilakukan ujicoba pada pembahasan poin-poin sebelumnya yaitu optimasi pemrograman web server, optimasi CSS (minify CSS), optimasi JavaScript (minify JavaScript) dan penggunaan AJAX. Dimana hasil pengujian dari faktor-faktor yang telah dituliskan sebelumnya akan ditampilkan dalam bentuk tabel, dimana pada Tabel 3 berikut akan dilihat perbandingan ukuran file, rasio kompresi, waktu akses awal dan waktu akses akhir.

Tabel 3. Ukuran Source File Pengujian Sebelum Optimasi dan Sesudah Optimasi

Nama file	Ukuran asli (bytes)	Ukuran Hasil Kompresi (bytes)	Rasio Kompresi (dalam %)
index.php	5215	4837	7.24
datauji_1.php	6487	5796	10.65
datauji_2.php	5899	5328	9.67
script1.js	58458	28456	51.32
script2.js	100196	46192	53.89
mos-style.css	18242	16295	10.67
style.css	5629	3686	34.51

Pembahasan berikutnya adalah yang merupakan inti dari topik penelitian ini yaitu mengenai kecepatan akses dan berbicara mengenai kecepatan akses disini adalah mengenai berapa lama waktu yang diperlukan untuk dapat membuka atau proses load halaman website seutuhnya. Dimana, untuk hasil pengujian ini juga akan ditampilkan dalam bentuk Tabel perbandingan waktu kecepatan akses untuk masing-masing halaman website sebelum dilakukan teknik optimasi dan sesudah dilakukan teknik optimasi.

Tabel 4. Pengukuran Hasil Pengujian Kecepatan Waktu Akses Halaman Website

Nama file	Jumlah Pengakses	Waktu Akses Awal (detik)	Waktu Akses Baru (detik)	Selisih Waktu (detik)
index.php	14	6.83	3.10	+3.73
datauji_1.php	14	5.32	4.48	+0.84
datauji_2.php	14	5.89	4.57	+1.32
script1.js	14	3.78	1.51	+2.27
script2.js	14	4.68	2.17	+2.51
mos-style.css	14	2.54	2.27	+0.27
style.css	14	1.79	1.58	+0.21

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 4, dapat dilihat bahwa terjadi peningkatan waktu akses yang lebih baik setelah dilakukan proses optimasi untuk masing-masing jenis file. Hal ini dapat dilihat dengan terdapat selisih waktu sebelum dilakukan optimasi dengan sesudah dilakukan optimasi.

Bagian terakhir adalah hasil pengujian halaman website dengan menggunakan bantuan tools online yaitu menggunakan gtmatrix untuk mengukur kinerja dari halaman website setelah dilakukan proses optimasi dan juga memberikan skor penilaian (Yslow Score) :



Gambar 5. Hasil Pengujian Halaman Website Menggunakan GTmetrix

Selain memberikan penilaian terhadap performa dari halaman website, tools gtmatrix juga memberikan informasi tambahan dari halaman website yang telah dilakukan proses scanning dan berikut adalah hasil halaman website yang telah dilakukan proses scanning pada pengujian penelitian ini, terutama pada bagian proses minify CSS dan juga kompresi file JavaScript :



Gambar 6. Hasil Penilaian Terhadap Beberapa Teknik Optimasi Halaman Website

IV. SIMPULAN

Kecepatan akses halaman website dipengaruhi oleh banyak faktor dan merupakan hal yang kompleks untuk dipelajari, karena faktor-faktor yang mempengaruhinya bisa terdiri dari banyak faktor yaitu dari sisi software, hardware, brainware dan bahkan media yang digunakan untuk mengakses halaman website itu sendiri. Oleh karena itu pada penelitian ini peneliti menyimpulkan beberapa hal yang didapatkan dari pengujian-pengujian yang telah

dilakukan selama proses penelitian ini dilakukan, yaitu :

1. Penggunaan teknik optimasi halaman website terutama dalam penggunaan metode minify CSS dan kompresi file JavaScript dapat mengurangi waktu tunggu (load) halaman website. Terutama halaman-halaman website yang memiliki ukuran besar.
2. Penggunaan teknologi AJAX pada penelitian ini membuktikan bahwa terjadinya pengurangan waktu secara signifikan untuk halaman website dalam memproses permintaan dari user.
3. Teknik penggabungan file JavaScript juga terbukti dapat mengurangi waktu tunggu dari proses load halaman website dan hal ini akan dapat menghemat penggunaan bandwidth, kemudian juga meringankan beban kerja dari server serta mempercepat proses transfer data dari server kepada klien.
4. Penggunaan beberapa tools untuk menguji performa dari waktu akses halaman website diperlukan untuk mendapatkan hasil yang lebih bervariasi, sehingga dapat memperlihatkan dengan lebih detail terhadap faktor-faktor tambahan yang dapat mempengaruhi kecepatan akses halaman website.

Optimasi yang dilakukan pada penelitian ini lebih berfokus pada teknik pemrograman web server (*server side programming*) dan juga *client side programming*, namun demikian penelitian-penelitian yang masih dapat dilakukan lebih lanjut untuk dapat melengkapi atau menambahkan teknik optimasi agar waktu akses halaman website meningkat juga dapat dilakukan dari faktor-faktor lainnya yaitu optimasi database server (query optimization), optimasi *images* yang digunakan, penggunaan teknik kompresi HTTP Requests. Selain itu hasil *benchmark* (pengukuran) dari optimasi performance dari halaman website juga dapat menggunakan beberapa web page tools analyzer untuk melihat seberapa besar tingkat signifikansi dari proses optimasi yang telah dilakukan untuk memperlihatkan perbedaan selisih kecepatan waktu akses halaman website.

DAFTAR PUSTAKA

- [1] A. Dhawan, K. Gaikwad, P. Gaikwad, and H. Kumbhar. 2015. "Journey of Internet – Past, Present & Future", *International Journal of Advanced Research in Computer Science Engineering and Information Technology*, Vol.5. No.2. hal.384-393.
- [2] N. Choudhury. 2014. "World Wide Web and Its Journey from Web 1.0 to Web 4.0". *International Journal of Computer Science and Information Technologies*, Vol.5 No.6, hal.8096-8100.
- [3] T. Crawford and T. Hussain. 2017. "A Comparison of Server Side Scripting Technologies". *Proceeding of International Conference Software Engineering Research and Practice (SERP) 2017*. Las Vegas, Nevada. July 17-20.
- [4] M. Hills, P. Klint, and J. Vinju. 2013. "An Empirical Study of PHP Feature Usage". *Proceeding of International Symposium on Software Testing and Analysis (ISSTA) 2013*, Lugano, Switzerland, July 15-20.
- [5] A. Håkansson. 2013. "Portal of Research Methods and Methodologies for Research Projects and Degree Projects". *The World Congress in Computer Science, Computer Engineering, and Applied Computing*, Las Vegas, Nevada, July 22-25.
- [6] N. Jöse, M. Buro, R. Elio, J. Hoover, I. Nikolaidis, M. Salavatipour, L. Stewart, and K. Wong. 2013. "About Computing Science Research Methodology". <https://webdocs.cs.ualberta.ca/~c603/readings/research-methods.pdf> diakses pada tanggal 29 Januari 2019.
- [7] J. Manhas. 2013. "A Study of Factors Affecting Websites Page Loading Speed for Efficient Web Performance". *Internasional Journal of Computer Sciences and Engineering (IJCSSE)*. Vol.1. No.3. hal.32-35.
- [8] Y. Sakamoto, S. Matsumoto, S. Tokunaga, S. Saiki, and M. Nakamura. 2015. "Empirical Study on Effects of Script Minification and HTTP Compression for Traffic Reduction". *Proceedings of 3rd International Conference on Digital Information, Networking, and Wireless Communications*, Moscow, Russia, Feb 3-5.
- [9] N. SyamimiSaid, R. Alsaqour, H. Shaker, M. Abdelhaq, O. Alsaqour, and M. Uddin. 2014. "Review on Web Performance". *Journal of Engineering and Applied Sciences*. Vol.9. No.1. hal.18-23.
- [10] O. Sawant and S. Godse. 2014. "Web-Page Complexity and Optimization Mechanism to Reduce Web-Page Load Time". *International Journal of Computing and Technology (IJCAT)*. Vol.1. No.9. hal.444-447.
- [11] Z. Munyaradzi, G. Maxmillan, and M.N. Amanda. 2013. "Effects of Web Page Contents on Load Time over the Internet". *International Journal of Science and Research (IJSR)*. Vol.2. No.9. hal.75-79.
- [12] E. Mjelde and A.L. Opdahl. 2017. "Load-Time Reduction Techniques For Device-Agnostic Web Sites". *Journal of Web Engineering*. Vol.16 No.3&4. Hal.311-346.
- [13] A. Mesbah. 2009. "Analysis and Testing of AJAX-based Single-page Web Applications". Phd Research Project Dissertation, Institute Programming Research and Algorithms (IPA), Delft University of Technology, Delft, Netherlands.
- [14] Himawan, I. J. Dewanto, and M. G. Rumara. 2017. "Penggunaan Teknologi AJAX (Asynchronous JavaScript and XML) Pada Aplikasi Website Penerimaan Mahasiswa Baru Universitas Matana". *Jurnal Penelitian SAINTEK*, Vol.22. No.2. hal 98-113.